EXHIBIT A

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of	Mansoor Alicherry et al.
For	OPTIMIZING LATENCIES IN CLOUD SYSTEMS BY INTELLIGENT COMPUTE NODE PLACEMENT
Serial Number	13/660,226
Filed	October 25, 2012
Art Unit	2199
Examiner	Qing Yuan Wu
Attorney Docket Number	ALC 3838
Confirmation Number	8560

AFTER FINAL AMENDMENT

Sir:

In response to the office action dated February 26, 2015, please amend the above-identified application as set forth below under the AFCP 2.0 program:

CLAIM AMENDMENTS begin on page 2 of this paper.

REMARKS begin on page 12 of this paper.

CLAIM AMENDMENTS

This listing of the claims will replace all prior versions and listings of claims in this application.

1. (Currently Amended) A method performed by a cloud controller for assigning compute nodes to data nodes, the method comprising:

obtaining, by the cloud controller, a set of compute nodes, a set of data nodes, and a set of edges between compute nodes and data nodes;

determining an assignment objective to be achieved in assigning compute nodes from the set of compute nodes to data nodes from the set of data nodes; and

~~applying an algorithm associated with the assignment objective to obtain a set of assignments based on a plurality of costs associated with the set of edges; and~~

assigning a compute node of the set of compute nodes to a data node of the set of data nodes based ~~on the set of assignments~~ on the assignment objective and a plurality of costs associated with the set of edges;

wherein the assignment objective is based on a latency cost used in obtaining a set of assignments by applying an algorithm.

2. (Original) The method of claim 1, wherein the assignment objective comprises minimizing a total latency and the algorithm comprises an assignment algorithm.

3. (Original) The method of claim 1, wherein the assignment objective comprises minimizing a maximum latency and the algorithm comprises:

iteratively performing a binary search to identify an optimum threshold value, wherein an iteration of the binary search comprises:

identifying a current threshold value,

generating a temporary set of edges that prevents selection of edges from the set of edges having a cost greater than the threshold value,

applying an assignment algorithm based on the temporary set of edges to generate a current solution, and

modifying selection of a next threshold value based on the current solution; and

returning a solution associated with the optimum threshold.

4. (Original) The method of claim 1, wherein the assignment objective comprises minimizing a total latency within a limit for a maximum latency and the algorithm comprises:

generating a modified set of edges that prevents selection of edges from the set of edges having a cost greater than the limit; and

applying an assignment algorithm based on the modified set of edges.

5. (Original) The method of claim 1, wherein the assignment objective comprises minimizing a maximum latency within a limit for a total latency and the algorithm comprises:

iteratively performing a binary search to identify an optimum threshold value, wherein an iteration of the binary search comprises:

identifying a current threshold value,

generating a temporary set of edges that prevents selection of edges from the set of edges having a cost greater than the threshold value,

applying an assignment algorithm based on the temporary set of edges to generate a current solution,

comparing a total latency of the current solution to the limit, and

modifying selection of a next threshold value based on the current solution and the comparison between the total latency of the current solution to the limit; and

returning a solution associated with the optimum threshold.

6. (Original) The method of claim 1, further comprising:

weighting an initial cost of an edge of the set of edges based on an amount of data to be processed from a data node associated with the edge to produce a weighted cost of the edge,

wherein the plurality of costs associated with the set of edges comprises the weighted cost of the edge.

7. (Original) The method of claim 1, further comprising:

duplicating a duplicated node, wherein the duplicated node comprises at least one of:

a compute node of the set of compute nodes, and

a data node of the set of data nodes,

whereby the set of assignments includes at least two assignments related to the duplicated node.

8. (Original) The method of claim 1, wherein the set of compute nodes comprises at least one established virtual machine (VM).

9. (Original) The method of claim 1, wherein the set of compute nodes comprises at least one virtual machine (VM) that has not yet been established, and

assigning a compute node of the set of compute nodes to a data node of the set of data nodes based on the set of assignments comprises establishing the VM.

10. (Original) The method of claim 1, further comprising:

obtaining a set of compute cliques, wherein a compute clique of the set of compute cliques comprises a set of compute nodes within a predetermined distance of each other,

wherein applying the algorithm comprises:

applying the algorithm for edges of the set of edges associated with compute nodes belonging to a first compute clique of the set of compute cliques to produce a first set of assignments,

applying the algorithm for edges of the set of edges associated with compute nodes belonging to a second compute clique of the set of compute cliques to produce a second set of assignments,

identifying a best set of assignments based on the first set of assignments and the second set of assignments, and

wherein assigning a compute node of the set of compute nodes to a data node of the set of data nodes based on the set of assignments comprises assigning a compute node of the set of compute nodes to a data node of the set of data nodes based on the identified best set of assignments.

11. (Original) The method of claim 10, wherein obtaining a set of compute cliques comprises:

adding a first compute node to a new compute clique;

adding a first plurality of compute nodes to the new compute clique, wherein the compute nodes of the first plurality of compute nodes are within a distance of half of a predetermined threshold to the first compute node; and

adding the new compute clique to the set of compute cliques.

12. (Original) The method of claim 11, further comprising:

before adding the new compute clique to the set of compute cliques, adding a second plurality of compute nodes to the new compute clique, wherein the compute nodes of the second plurality of compute nodes are within a distance of the predetermined threshold to the first compute node and the first plurality of compute nodes.

13. (Currently Amended) A cloud controller for assigning compute nodes to data nodes, the cloud controller comprising:

a memory; and

a processor in communication with the memory, the processor being configured to:

obtain a set of compute nodes, a set of data nodes, and a set of edges between compute nodes and data nodes,

determine an assignment objective to be achieved in assigning compute nodes from the set of compute nodes to data nodes from the set of data nodes,

~~apply an algorithm associated with the assignment objective to obtain a set of assignments based on a plurality of costs associated with the set of edges, and~~

assign a compute node of the set of compute nodes to a data node of the set of data nodes based ~~on the set of assignments~~ on the assignment objective and a plurality of costs associated with the set of edges;

wherein the assignment objective is based on a latency cost used in obtaining a set of assignments by applying an algorithm.

14. (Original) The cloud controller of claim 13, wherein the assignment objective comprises minimizing a total latency and the algorithm comprises an assignment algorithm.

15. (Original) The cloud controller of claim 13, wherein the assignment objective comprises minimizing a maximum latency and the algorithm comprises:

iteratively performing a binary search to identify an optimum threshold value, wherein an iteration of the binary search comprises:

identifying a current threshold value,

generating a temporary set of edges that prevents selection of edges from the set of edges having a cost greater than the threshold value,

applying an assignment algorithm based on the temporary set of edges to generate a current solution, and

modifying selection of a next threshold value based on the current solution; and

returning a solution associated with the optimum threshold.

16. (Original) The cloud controller of claim 13, wherein the assignment objective comprises minimizing a total latency within a limit for a maximum latency and the algorithm comprises:

generating a modified set of edges that prevents selection of edges from the set of edges having a cost greater than the limit; and

applying an assignment algorithm based on the modified set of edges.

17. (Original) The cloud controller of claim 13, wherein the assignment objective comprises minimizing a maximum latency within a limit for a total latency and the algorithm comprises:

iteratively performing a binary search to identify an optimum threshold value, wherein an iteration of the binary search comprises:

identifying a current threshold value,

generating a temporary set of edges that prevents selection of edges from the set of edges having a cost greater than the threshold value,

applying an assignment algorithm based on the temporary set of edges to generate a current solution,

comparing a total latency of the current solution to the limit, and

modifying selection of a next threshold value based on the current solution and the comparison between the total latency of the current solution to the limit; and

returning a solution associated with the optimum threshold.

18. (Original) The cloud controller of claim 13, wherein the processor is further configured to:

weight an initial cost of an edge of the set of edges based on an amount of data to be processed from a data node associated with the edge to produce a weighted cost of the edge,

wherein the plurality of costs associated with the set of edges comprises the weighted cost of the edge.

19. (Original) The cloud controller of claim 13, wherein the processor is further configured to:

duplicate a duplicated node, wherein the duplicated node comprises at least one of:

a compute node of the set of compute nodes, and

a data node of the set of data nodes,

whereby the set of assignments includes at least two assignments related to the duplicated node.

20. (Original) The cloud controller of claim 13, wherein the set of compute nodes comprises at least one established virtual machine (VM).

21. (Original) The cloud controller of claim 13, wherein the set of compute nodes comprises at least one virtual machine (VM) that has not yet been established, and

in assigning a compute node of the set of compute nodes to a data node of the set of data nodes based on the set of assignments, the processor is configured to establish the VM.

22. (Original) The cloud controller of claim 13, wherein the processor is further configured to:

obtain a set of compute cliques, wherein a compute clique of the set of compute cliques comprises a set of compute nodes within a predetermined distance of each other,

wherein, in applying the algorithm, the processor is configured to:

apply the algorithm for edges of the set of edges associated with compute nodes belonging to a first compute clique of the set of compute cliques to produce a first set of assignments,

apply the algorithm for edges of the set of edges associated with compute nodes belonging to a second compute clique of the set of compute cliques to produce a second set of assignments, and

identify a best set of assignments based on the first set of assignments and the second set of assignments, and

wherein, in assigning a compute node of the set of compute nodes to a data node of the set of data nodes based on the set of assignments, the processor is configured to assign a compute node of the set of compute nodes to a data node of the set of data nodes based on the identified best set of assignments.

23. (Original) The cloud controller of claim 22, wherein, in obtaining a set of compute cliques, the processor is configured to:

add a first compute node to a new compute clique;

add a first plurality of compute nodes to the new compute clique, wherein the compute nodes of the first plurality of compute nodes are within a distance of half of a predetermined threshold to the first compute node; and

add the new compute clique to the set of compute cliques.

24. (Original) The cloud controller of claim 23, wherein the processor is further configured to: before adding the new compute clique to the set of compute cliques, add a second plurality of compute nodes to the new compute clique, wherein the compute nodes of the second plurality of compute nodes are within a distance of the predetermined threshold to the first compute node and the first plurality of compute nodes.

REMARKS

Claims 1-24 are pending in this application, of which claims 1 and 13 are independent. By this amendment, claims 1 and 13 are amended. Please reconsider the amended claims under the AFCP 2.0 program. No new matter is added.

Entry of the amendments is proper under 37 CFR 1.116 since the amendments: (a) place the application in condition for allowance (for the reasons discussed herein); (b) do not raise any new issues requiring further search and/or consideration (because the amendments amplify issues previously discussed throughout the prosecution); (c) satisfy a requirement of form asserted in the previous Office Action; (d) do not present any additional claims without canceling a corresponding number of finally rejected claims; and/or (e) place the application in better form for appeal, should an appeal be necessary.

ALLOWABLE SUBJECT MATTER

The Applicant thanks the Examiner for the indication of claims 3-7, 10-12, 15-19, and 22-24 as reciting allowable subject matter. The Applicant declines to rewrite any of these claims in independent form at this time, however, because the remaining claims are allowable over the cited references for the reasons presented below.

REJECTIONS UNDER 35 U.S.C. §§ 102 AND 103

The office action rejects claims 1, 8-9, 13, and 20-21 under 35 U.S.C. § 102 as allegedly being anticipated by U.S. patent application publication number 2013/0318525 (hereinafter, "Palanisamy et al.").

The office action rejects claims 2 and 14 under 35 U.S.C. § 103 as allegedly being unpatentable over Palanisamy (and no secondary references).

The Applicant respectfully traverses these rejections. Claim 1 (as amended) recites,

> A method performed by a cloud controller for assigning compute nodes to data nodes, the method comprising:
> obtaining, by the cloud controller, a set of compute nodes, a set of data nodes, and a set of edges between compute nodes and data nodes;
> *determining an assignment objective to be achieved in assigning compute nodes from the set of compute nodes to data nodes from the set of data nodes*; and
> assigning a compute node of the set of compute nodes to a data node of the set of data nodes based on the assignment objective and a plurality of costs associated with the set of edges;
> *wherein the assignment objective is based on a latency cost used in obtaining a set of assignments by applying an algorithm.*

(emphasis added). Claim 13 includes language that is similar to the above-emphasized subject matter.

In the final office action, Examiner argues,

> The examiner respectfully disagrees and submits that **applicant's argument is directed to additional limitations** that are recited in subsequent dependent claims that are separately addressed. As recited in claim 1, the **"determining" of "an assignment objective to be achieved" can be broadly interpreted as the desire or intended result of *assignment of compute nodes to data nodes***, such that Palanisamy clearly teaches an objective, that is determined, to lower data transfer network hop and distance of data access to analyze data "efficiently", "quickly" and the intention of reducing network traffic by different assignment/placement of data and/or VMs on different nodes [abstract; paragraphs 2-4 and 221, therefore applicant's argument is not persuasive.
> More specifically, the different assignment objective(s) to be achieved by different algorithm(s) as intended by applicant is not recited in claim 1. Furthermore, in response to applicant's argument that the reference(s) fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., VM placement) are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See In re Van Geuns, 988 F.2d 118 1,26 USPQ2d 1057 (Fed. Cir. 1993). The limitation "assigning" can be broadly interpreted as a designation or association of compute nodes and data nodes and in no way implied the "placement" or temporal relationship of the occurrence of the "placement" of nodes as argued by applicant. Therefore applicant's argument is not persuasive.

This rejection is in error because the office action employs an unreasonably broad interpretation of the claims, taking the position that the recited "determining an assignment objective" encompasses any intention indicated in Palnisamy. Specifically, the office action argues that the assignment objective is "to lower data transfer network hop and distance of data access to analyze data "efficiently", "quickly" and the intention of reducing network traffic."

This definition of the phrase "determining an assignment objective" is unreasonably broad. Examiner is respectfully, not entitled to simply choose definitions to support his argument. The definition disclosed must be "the ordinary and customary meaning given to the term by those of ordinary skill in the art" and also must be "consistent with the specification." M.P.E.P. § 2111; 2111.01(I). The person of ordinary skill in the art would not have given the term a meaning so broad as any motivation as proposed by Examiner.

Further, such an interpretation would be wholly inconsistent with the specification; a reading of an "assignment objective" as any network node is unreasonable in view of the specification's various disclosures such as "minimizing a total latency, minimizing a maximum latency, minimizing a total latency within a limit on maximum latency, and minimizing a maximum latency within a limit on total latency" Present Application, paragraph [0044]. For at least these reasons, Examiner's interpretation is in contradiction with the M.P.E.P., and, thus, independent claims 1 and 13 are allowable over the references of record.

When the claim is taken as a whole, examiners argument simply does not make sense. Claim 1 reads,

> obtaining, by the cloud controller, a set of compute nodes, a set of data nodes, and a set of edges between compute nodes and data nodes;

determining an assignment objective to be achieved in assigning compute nodes from the set of compute nodes to data nodes from the set of data nodes;

Determining here, is an action required in the claim. A tertiary mention of a motivation to assign one type of node to another in Palanisamy, does not by any means imply an action step which must be taken in a sequence as required by the claim.

Further, examiner has argued that the dependent claim limitations were being argued in the non-final office action response. This is respectfully not the case. Examples were given to illustrate what examiner has not accurately cited. The office action cites paragraphs [0045-48] of Palanisamy as allegedly disclosing this subject matter. These paragraphs, however, disclose only that virtual machines (VM) will be placed on the same nodes as the data sets on which the VMs will operate. There is no disclosure of a separate step of determining an assignment objective. The system of *Palanisamy always follows the same approach to VM assignment: collocate the VM with the data if possible, otherwise minimize hops.* (Palanisamy, paragraph [0064-65]). There is no opportunity to determine which assignment objective is to be achieved and apply that algorithm associated with the selected objective because the single approach appears to simply be hard-coded into the system.

For the foregoing reasons, Palanisamy fails to disclose, *inter alia*, "determining an assignment objective to be achieved in assigning compute nodes from the set of compute nodes to data nodes from the set of data nodes," and therefore fails to disclose each and every element recited by the claims. Claims 1 and 13 are therefore allowable over the cited reference. Claims 2 and 8-9 depend from allowable claim 1; and claims 14 and 20-21 depend from allowable claim 13. These claims are therefore allowable based on their respective dependencies, as well as the separately

patentable subject matter recited therein. In view of the foregoing, the Applicant respectfully requests that the rejections under 35 U.S.C. §§ 102 and 103 be withdrawn.

CONCLUSION

While the Applicant believes that the application is currently in condition for allowance, as demonstrated by the above remarks, should the Examiner have any further comments or suggestions, it is respectfully requested that the Examiner telephone the undersigned attorney in order to expeditiously resolve any outstanding issues.

In the event that the fees submitted prove to be insufficient in connection with the filing of this paper, please charge our Deposit Account Number 50-0578 and please credit any excess fees to such Deposit Account.

Respectfully submitted,
KRAMER & AMADO, P.C.

Date: April 13, 2015

/ Terry W. Kramer/
Terry W. Kramer
Registration No.: 41,541

KRAMER & AMADO, P.C.
330 John Carlyle Street, 3rd Floor
Alexandria, VA 22314
Phone: 703-519-9801
Fax: 703-519-9802

Doc Code: A.NE.AFCP
Document Description: After Final Consideration Pilot Program Request

PTO/SB/434 (05-13)

CERTIFICATION AND REQUEST FOR CONSIDERATION UNDER THE AFTER FINAL CONSIDERATION PILOT PROGRAM 2.0

Practitioner Docket No.:	Application No.:	Filing Date:
ALC 3838	13/660,226	October 25, 2012
First Named Inventor:	**Title:**	
Mansoor Alicherry et al.	OPTIMIZING LATENCIES IN CLOUD SYSTEMS BY INTELLIGENT COMPUTE NODE PLACEMENT	

APPLICANT HEREBY CERTIFIES THE FOLLOWING AND REQUESTS CONSIDERATION UNDER THE AFTER FINAL CONSIDERATION PILOT PROGRAM 2.0 (AFCP 2.0) OF THE ACCOMPANYING RESPONSE UNDER 37 CFR 1.116.

1. The above-identified application is (i) an original utility, plant, or design nonprovisional application filed under 35 U.S.C. 111(a) [a continuing application (*e.g.*, a continuation or divisional application) is filed under 35 U.S.C. 111(a) and is eligible under (i)], or (ii) an international application that has entered the national stage in compliance with 35 U.S.C. 371(c).
2. The above-identified application contains an outstanding final rejection.
3. Submitted herewith is a response under 37 CFR 1.116 to the outstanding final rejection. The response includes an amendment to at least one independent claim, and the amendment does not broaden the scope of the independent claim in any aspect.
4. This certification and request for consideration under AFCP 2.0 is the only AFCP 2.0 certification and request filed in response to the outstanding final rejection.
5. Applicant is willing and available to participate in any interview requested by the examiner concerning the present response.
6. This certification and request is being filed electronically using the Office's electronic filing system (EFS-Web).
7. Any fees that would be necessary consistent with current practice concerning responses after final rejection under 37 CFR 1.116, *e.g.*, extension of time fees, are being concurrently filed herewith. [There is no additional fee required to request consideration under AFCP 2.0.]
8. By filing this certification and request, applicant acknowledges the following:
    - Reissue applications and reexamination proceedings are not eligible to participate in AFCP 2.0.
    - The examiner will verify that the AFCP 2.0 submission is compliant, *i.e.*, that the requirements of the program have been met (see items 1 to 7 above). For compliant submissions:
        - The examiner will review the response under 37 CFR 1.116 to determine if additional search and/or consideration (i) is necessitated by the amendment and (ii) could be completed within the time allotted under AFCP 2.0. If additional search and/or consideration is required but cannot be completed within the allotted time, the examiner will process the submission consistent with current practice concerning responses after final rejection under 37 CFR 1.116, *e.g.*, by mailing an advisory action.
        - If the examiner determines that the amendment does not necessitate additional search and/or consideration, or if the examiner determines that additional search and/or consideration is required and could be completed within the allotted time, then the examiner will consider whether the amendment places the application in condition for allowance (after completing the additional search and/or consideration, if required). If the examiner determines that the amendment does not place the application in condition for allowance, then the examiner will contact the applicant and request an interview.
            - The interview will be conducted by the examiner, and if the examiner does not have negotiation authority, a primary examiner and/or supervisory patent examiner will also participate.
            - If the applicant declines the interview, or if the interview cannot be scheduled within ten (10) calendar days from the date that the examiner first contacts the applicant, then the examiner will proceed consistent with current practice concerning responses after final rejection under 37 CFR 1.116.

Signature	Date
/Terry W. Kramer/	2015-04-13
Name (Print/Typed)	**Practitioner Registration No.**
Terry W. Kramer	41541

***Note**: This form must be signed in accordance with 37 CFR 1.33. See 37 CFR 1.4(d) for signature requirements and certifications. Submit multiple forms if more than one signature is required, see below*.*

☐ * Total of _____ forms are submitted.